Amendments to the Drawings:

The attached replacement sheets of drawings includes changes to Figs. 1, 2 and 3 and replaces the original sheets including Figs.1, 2 and 3.

In Figure 1, the applicant changed the indicated name of the component marked 26a to "SDRAM Controller", and changed the indicated name of the component marked 26b to "SRAM Controller." Additionally, applicant has provided legible markings for the various components appearing in the figure that were deemed to have illegible markings.

In Figure 2, the applicant presents the schematic shown in old FIG. 2 in two separate drawing sheets, Figure 2A and Figure 2B. The two replacement figures show more clearly the details shown in old Figure. 2, and conform to the statutory requirements of 37 C.F.R. § 1.84.

In Figure 3, the applicant marked the program counter units uPC-1 to uPC-4 with their corresponding reference numerals. Thus, uPC-1 is marked with reference numeral 72a (which previously pointed to uPC-4), uPC-2 is marked with reference numeral 72b (which previously pointed to uPC-1), and uPC-3 is marked with reference numeral 72c (which previously pointed to the illustrated multiplexer). Additionally, new reference numeral 72d was added to mark uPC-4.

Attachments following last page of this Amendment:

Replacement Sheet (6 pages)
Annotated Sheet Showing Change(s) (3 pages)

REMARKS

Applicant amended the specification to make the minor corrections on pages 3 and 10 requested by the examiner. In addition, applicant amended the paragraph beginning on page 3, line 4 to indicate that the component marked 16c in FIG. 1 is a Flash ROM unit that can be accessed by the microengines.

Additionally, applicant amended the paragraph beginning at page 11, line 6, to clarify that the registers BP2-BP0 are breakpoint registers. Support for this clarification is provided by claims 7 and 26 of the originally filed application.

Applicant also amended the abstract, as requested by the examiner, to clarify that the method of operating the processor performs direct write operations to the processor's registers, including, for example the processor's control and status registers. Applicant further amended the abstract to clarify that the disclosed method includes the feature of loading data to selected bits according to the processing thread number.

No new matter has been added to the specification.

The examiner objected to the drawings under 37 C.F.R. §1.83(a) on the ground that the provisions of 37 C.F.R. §1.83(a) require that the drawings show every feature of the invention, and that therefore the subject matter of claims 1-5, 7-24, and 26-38 must be shown in the drawings, or the feature(s) cancelled from the claims.

Applicant traverses the examiner's objections to the drawings under 37 C.F.R. §1.183(a). Applicant contends that Applicant's figures comply with 37 C.F.R. §1.83(a).

35 U.S.C. §113 states "[t]he applicant shall furnish a drawing where necessary for the understanding of the subject matter to be patented." Further, as provided by 37 C.F.R. §1.81(a), "[t]he applicant for a patent is required to furnish a drawing of his or her invention where necessary for the understanding of the subject matter sought to be patented."

Thus, contrary to the examiner's position, the main statutory provisions of the Patent Act regarding drawings, as well as rule 1.81(a), make it clear that it is not necessary to show in the drawings every feature of the claims as specified in the claims. Rather, it is only when the

subject matter in the claims cannot be understood without the assistance of drawings that drawings become necessary.

Additionally, in discussing the requirements of 37 C.F.R. §1.83(a), MPEP 608.02(d) notes "[a]ny structural detail that is sufficient importance to be described should be shown in the drawings" (emphasis added). Thus, the Patent Office's own interpretation of the provisions of 37 C.F.R. §1.83(a) makes it clear that furnishing drawings corresponding to the features in the claims is not a mandatory requirement (as indicated by the use of the discretionary language "should be shown" in MPEP 608.02(d)).

Nevertheless, the drawings comply with 37 C.F.R. §1.83(a). Independent claim 1 is directed to a method for operating a processor. FIG. 1 shows a processor. The method includes receiving data, and then loading data into selected bits of a register according to the number of the processing thread. Thus, the details of the elements of claim 1 are sufficiently clear that they can be understood without the use of drawings. Applicant also notes that page 10, lines 25-26, of the originally filed application shows an embodiment of an instruction that causes the processor to perform the method. The applicant, therefore, submits that it is not necessary, in this case, to have a drawing illustrating these features of independent claim 1.

Claims 2-5, and 7-19 describe additional features of the method described in claim 1, including, for example, specifying how "loading" is performed (see claim 7), the format of a token used in performing the operation described in claims 1 (see claims 8-19), etc. Applicant notes that the table at page 11, lines 20, to page 12, line 10, of the originally filed application shows the options and parameters associated with the method. Thus, the features of claim 2-5, and 7-19 are readily understood without the use of drawings.

Claims 20-38, which recite features similar to those recited in claims 1-19, can likewise be understood without drawings.

The examiner objected to claims 3, 5, 7, 11, 22, 24, 30 and 38. Applicant amended these claims in accordance with the examiner's comments. Applicant thanks the examiner for pointing out the various typographical and grammatical errors the examiner noted.

Additionally, claim 37 was amended for greater clarity.

The examiner rejected claims 7 and 26 under 35 U.S.C. §112, 1st paragraph on the ground that it does not appear from the claims that, at the time the application was filed, the applicant had possession of the claimed invention. In particular, the examiner was unclear, for example, on whether the shifting operation described in claim 7 is performed before or after the loading operation recited in claim 1.

Applicant amended claim 7 to correct a typographical error by replacing the wording "shifting comprises" that appeared in the preamble with "wherein loading the data comprises". Applicant similarly amended claim 26. Thus, amended claims 7 and 26 provide that in some embodiments loading the data includes shifting portions of the data. Support for this feature is provided, for example, in page 11, lines 6-9 of the originally filed application. Applicant believes that with this amendment the relationship between the "loading" and "shifting" operations recited in claims 7 and 26 is clearly delineated. Applicant thus traverses the examiner's rejections under 35 U.S.C. §112, 1st paragraph.

The examiner rejected claims 3 and 22 under 35 U.S.C. §112, 2nd paragraph on the ground that it is not clear how registers can be contained in a bus (as previously recited in claims 3 and 22).

In response, applicant amended claims 3 and 22 to correct the wording "in contained in" to "is coupled to," clarifying the structural relationship between the registers and the bus.

The examiner also rejected claims 7 and 26 under 35 U.S.C. §112, 2nd paragraph on the ground that it is not clear what shifting data into a "BP register 2 to a BP register 0" means.

In response, the applicant amended claims 7 and 26 to clarify that data is shifted into bits in the register corresponding to breakpoint register BP2 through breakpoint register BP0. Support for this clarification is provided in page 11, lines 6-9 of the originally filed application.

Thus, claims 7 and 26 describe that a second portion of the data is shifted into bits that correspond to breakpoint registers. In some embodiments applicant's processor architecture may include breakpoint registers that facilitate debugging procedures. Control of such breakpoint

registers may be performed in some embodiments using control and status registers (such as those described in applicant's claims 2, and 21). Applicant's claims 7 and 26 thus describe, for example, that in such embodiments data to control the operations of the breakpoint registers is shifted into bit positions (in the respective control and status registers) that correspond to the breakpoint registers.

Applicant amended independent claim 1 to clarify that data is loaded into selected bits of a register according to the processing thread number. Support for this feature is found, for example, at page 10, lines 17-24 of the originally filed specification. Applicant similarly amended independent claim 20.

The examiner rejected claims 1-6, 8-10, 20-25 and 27-29 under 35 U.S.C. §102(e) as being anticipated by U.S. Patent No. 6,058,465 to Nguyen.

The examiner also rejected claims 1-6, 8-9, 20-25 and 27-28 under 35 U.S.C. §102(b) as being anticipated by "IA-64 Application Developer's Architecture Guide", May 1999 (hereinafter "Intel").

Additionally, the examiner rejected claims 11-19 and 30-38 under 35 U.S.C. §103(a) as being unpatentable over Nguyen, and further rejected claims 7, 10-19, 26, and 29-38 under 35 U.S.C. §103(a) as being unpatentable over Intel.

Applicant's independent claim 1 recites a method of operating a processor that includes "loading [received] data into selected bits of a register according to the processing thread number." Applicant's method enables changing of the content of control registers by setting values of particular bits that correspond to particular threads.

Nguyen describes a vector processor architecture having vector registers, and corresponding instructions to manipulate operands associated with the vector registers (Abstract). Nguyen's processing architecture uses a vector processor 120 that includes a control register, called VCSR, to control the operation of the processor, and more generally the operation of the processing core 100 (see, for example, col. 5, lines 9-17, and Tables C.4 and C.5 at cols. 27 and 28).

As shown in Table C.5, the VCSR register is a 32-bit register whose various bits enable selection of various processing modes and options. For example, the CBANK bit indicates which of the processor's two 32-vector register banks are to be used. If the CBANK is set to 1, bank 1 of the vector-registers is selected, whereas if the CBANK bit is cleared (i.e., it is 0), vector-register bank 0 is selected (col. 5, lines 9-12, and col. 28, lines 16-18). As Nguyen discloses, a user can select particular modes of processor operation using Change Vector Register Instruction (VCHGCR) to set those bits that correspond to the desired modes of operation (col. 91-92). Nowhere, however, does Nguyen disclose that any bits of the VCSR register correspond to particular threads. Therefore, Nguyen neither discloses nor suggests loading data, using the VCHGCR instruction or otherwise, to selected bits according to a processing thread number, as required by applicant's independent claim 1.

Intel describes an instruction set for the IA-64 processing architecture. One instruction used for in IA-64 architecture is the Shift Left ("*shl*") instruction. As explained in Intel's page 7-165, execution of the *"shl"* instruction causes the value stored in source register r_2 to be left shifted by an amount specified by register r_3, or alternatively by a numerical value specified as a parameter of the instruction. Bit positions vacated by the shifted value in register r_2 are filled with zeros. The shifted value in register r_2 is then loaded into a specified target register r_1 (as indicated by one of the parameters of the *shl* instruction). However, nowhere does Intel disclose that data is loaded into selected bits of any register according to a processing thread number.

Although the IA-64 supports multiple thread execution, the use of threads does not affect the operation of the *shl*, or for that matter the other instructions referred to by the examiner on Intel's page C-23. That is, the particular processing thread number does not determine which bits of any of the registers used in the execution of the *shl* instruction are loaded with data. Thus, Intel does not disclose or suggest "loading the data into selected bits of a register according to the processing thread number," as required by applicant's independent claim 1.

Since neither Nguyen nor Intel discloses or suggests, alone or in combination at least the feature of "loading the data into selected bits of a register according to the processing thread number," applicant's independent claim 1 is patentable over the art cited by the examiner.

Claims 2-19 depend from independent claim 1 and are therefore patentable for at least the same reasons as independent claim 1.

Independent claim 20 recites the feature of "load the data into selected bits of a register according to the processing thread number." For reasons similar to those provided with respect to independent claim 1, at least this feature is not disclosed by the cited art. Accordingly, independent claim 20 is patentable over the cited art. Claims 21-38 depend from independent claim 20 and are therefore patentable for at least the same reasons as independent claim 20.

Further, as noted above, the examiner rejected claim 7 and 26 under 35 U.S.C. §103(a) as being unpatentable over Intel.

Amended claim 7 describes "shifting a second portion of the data into bits corresponding to a breakpoint (BP) register 2 through BP register 0."

While, as explained above, Intel's *shl* instruction loads a destination register with the shifted content of a source destination, nowhere does Intel describe that the either of the source register or the destination register has bits that correspond to breakpoint registers. Thus, Intel does not disclose or suggest "shifting a second portion of the data into bits corresponding to a breakpoint (BP) register 2 through BP register 0," as required by applicant's claim 7.

Similarly, Nguyen also does not disclose or suggest that any of the bits on the VCSR register correspond to breakpoint registers. Nguyen, therefore, does not disclose or suggest "shifting a second portion of the data into bits corresponding to a breakpoint (BP) register 2 through BP register 0," as required by applicant's claim 7.

Since none of the references cited by the examiner discloses at least the feature "shifting a second portion of the data into bits corresponding to a breakpoint (BP) register 2 through BP register 0," applicant's claim 7 is patentable over the cited art.

Claim 26 recites "shift a second portion of the data into bits corresponding to a breakpoint (BP) register 2 through BP register 0." For similar reasons as those provided with respect to claim 7, at least this feature is not disclosed by the art. Accordingly, applicant's claim 26 is patentable over the cited art.

The examiner also rejected claims 18 and 37 under 35 U.S.C. §103(a) as being unpatentable over Nguyen, and also as being unpatentable over Intel.

Claim 18 recites "wherein a context (CTX) field overrides a default context if bit 2 is set." Thus, if bit 2 is set, the loading operation of applicant's method will be based on a context other than the context of the processing thread (for example, the context may be determined by the context number specified by bits 0:1, as provided in claim 19).

The examiner stated: "Nguyen has further taught that a context (CTX) field overrides a default context is [*sic*] bit 2 is set. From columns 91-92, if bit 2 is set, the current SMM bit operation is changed from whatever it currently is to either clear or toggle operation" (paragraph 56, page 17 of the Office Action). The applicant disagrees.

As explained above, Nguyen's VCSR register is a 32-bit register whose various bits enable selection of various processing modes and options. For example, Nguyen describes that bits 2:3 of the VCHGCR instruction control the VCSR<SMM> bit. Nguyen further discloses at Table C.5 that the SMM bit is the Select Move Mask, and that "[w]hen this bit is set, the VMMR0/1 pair becomes the element mask for the arithmetic operations." Thus, bits 2-3 of the VCHGCR instruction affect the selection of a mask used for arithmetic operation. Bits 2-3 of the VCHGCR instruction, therefore, do not affect in anyway the use and/or selection of context, and they certainly do not cause or affect overrides of a default context, as provided by applicant's claim 18.

The examiner also stated that: "Intel has further taught that a context (CTX) field overrides a default context is [*sic*] bit 2 is set. See page C-4 and note that if bit 2 is set, a certain qualifying predicate register is addressed. If the qualifying predicate is set to 0, then the instruction will not be executed, when the default is to execute anything that is selected for execution" (paragraph 69, page 21 of the Office Action). Applicant respectfully disagrees.

Applicant first notes that it is not entirely clear, which bit 2, in the "Instruction Format Summary" on page C-4, the examiner is referring to. Indeed, it appears from the table on page C-4 that bit 2 for any of the instructions is a reserved bit.

In any event, applicant notes that none of the bits of the instructions described in Intel, or referred to by the examiner, has anything to do with contexts, and nowhere does Intel disclose or suggest a bit, which if set, causes a default context to be overridden, as provided by applicant's claim 18.

Since neither Nguyen nor Intel discloses or suggests, alone or in combination, at least the feature of "wherein a context (CTX) field overrides a default context if bit 2 is set," claim 18 is therefore patentable over the cited art.

Claim 37 recites "wherein a context (CTX) field overrides a default context if bit 2 is set." For similar reasons as those provided with respect to claim 18, at least this feature is not disclosed by the cited art. Accordingly, claim 37 is patentable over the cited art.

In view of the foregoing remarks, applicant respectfully submits that the application is in condition for allowance and such action is respectfully requested at the examiner's earliest convenience.

It is believed that all the rejections and/or objections raised by the examiner have been addressed.

All of the dependent claims are patentable for at least the reasons for which the claims on which they depend are patentable.

Canceled claims, if any, have been canceled without prejudice or disclaimer.

Any circumstance in which the applicant has (a) addressed certain comments of the examiner does not mean that the applicant concedes other comments of the examiner, (b) made arguments for the patentability of some claims does not mean that there are not other good reasons for patentability of those claims and other claims, or (c) amended or canceled a claim does not mean that the applicant concedes any of the examiner's positions with respect to that claim or other claims.

No fee is believed due. Please apply any charges or credits to deposit account 06-1050, referencing attorney docket 10559-308US1.

Respectfully submitted,

Date: Nov. 30, 2005

Ido Rabinovitch
Attorney for Intel Corporation
Reg. No. L0080

Fish & Richardson P.C.
225 Franklin Street
Boston, MA 02110
Telephone: (617) 542-5070
Facsimile: (617) 542-8906

21216028.doc

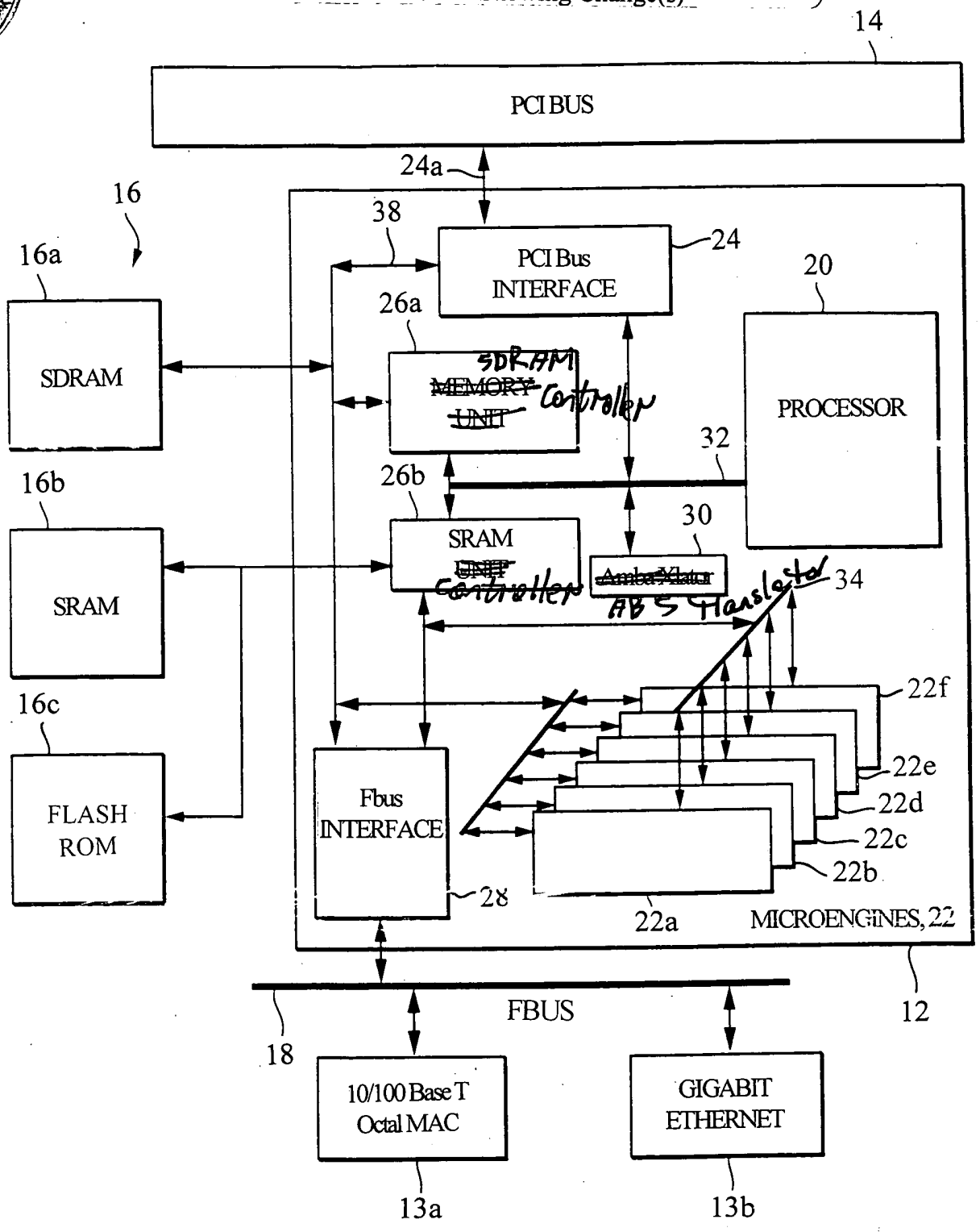
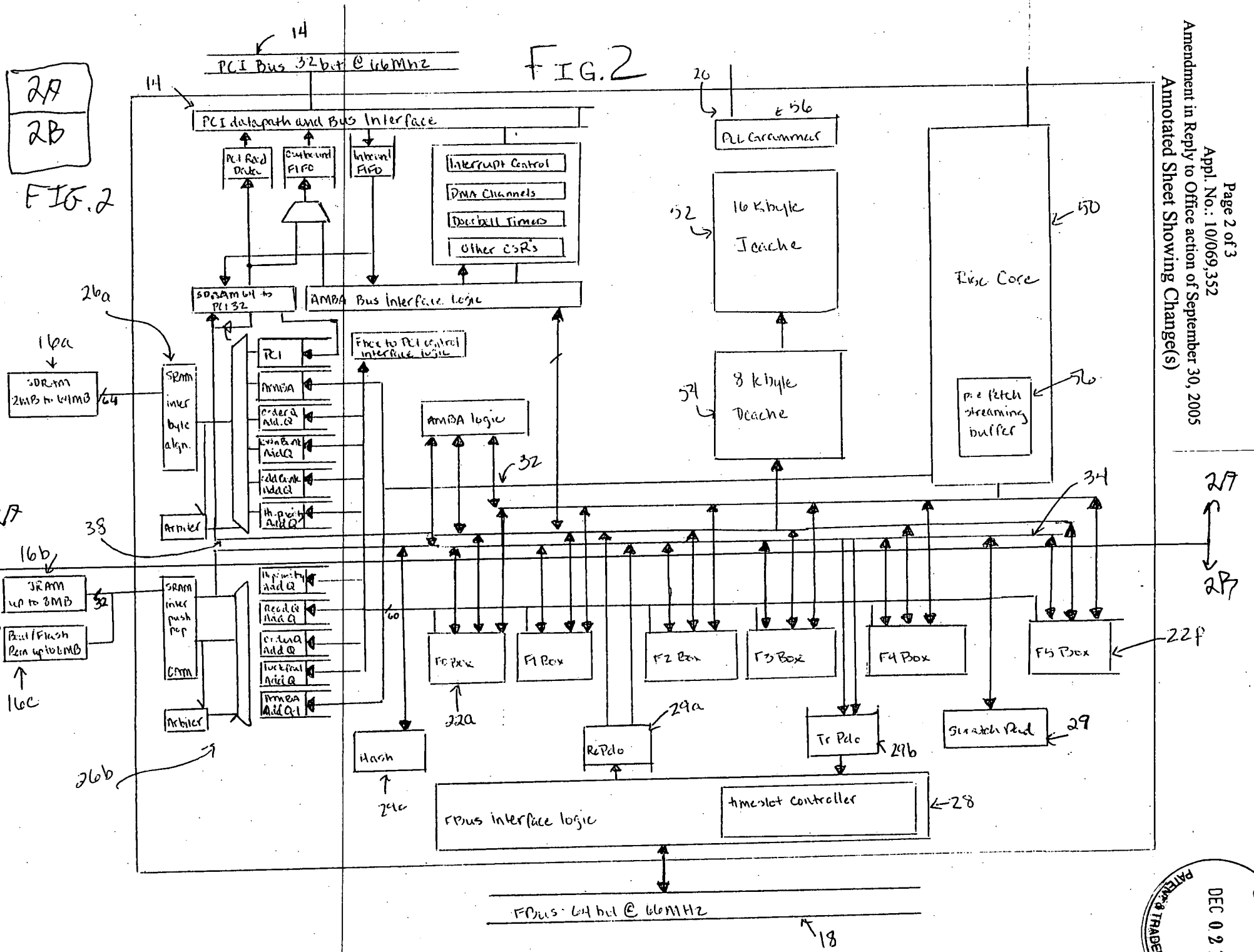


FIG. 1

FIG. 2



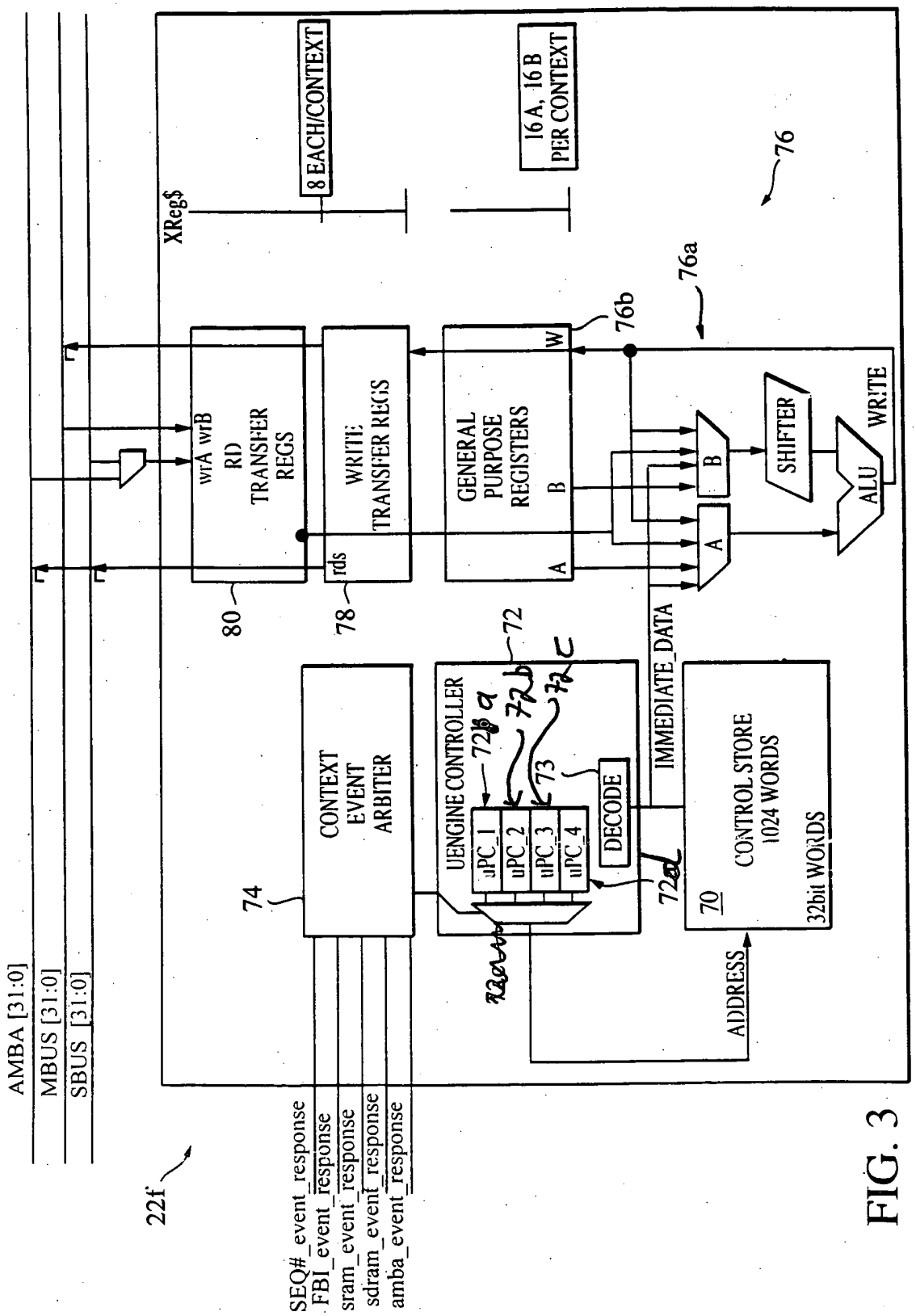


FIG. 3